**Prueba técnica de Sipecom, en JAVA.**

**Lineamientos técnicos:**

1. Esta prueba, sirve para medir el conocimiento técnico en el uso de herramientas JAVA, PHP y SQL Server.
2. La prueba consiste en la creación de dos aplicaciones, la primera es un desarrollo en JAVA JDK 1.8 y la segunda en PHP 5.6.
3. El desarrollo en JAVA debe mantener técnicas de Programación Orientada a Objetos, se recomienda el uso de un esquema de 3 capas, uno de base de datos que puede ser usando un ORM o métodos directos de conexión, una capa de servicios y una de presentación.
4. En la estructura del código en PHP se evaluara el uso de buenas prácticas de desarrollo y programación ordenada.

**Requerimiento funcional:**

**Aplicativo en PHP**

Para el aplicativo PHP debes 2 ventanas.

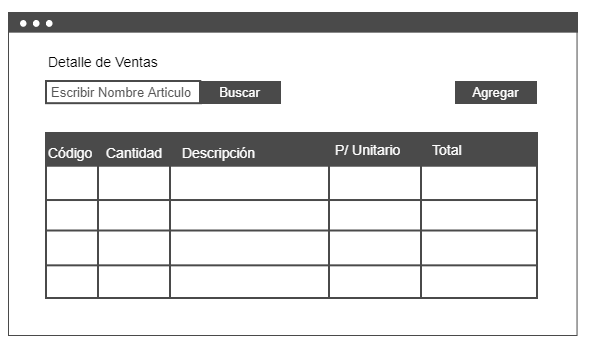
La primera ventana “Autenticación” tendrá un formulario de login, con los campos usuario y contraseña cifrada. Para validar los campos usa la tabla login\_users

Los campos de la tabla login\_users son:

|  |  |  |
| --- | --- | --- |
| **Campo** | **Tipo de Dato** | **Observación** |
| username | Varchar(50) |  |
| contraseña | Varchar(100) |  |

Para ingresar al sistema, deberá funcionar con el usuario “sipecom” y la contraseña “Sipecom2020.” que estará ingresado previamente en la base de datos. La contraseña se encontrara cifrada con el password\_hash default de PHP.

Al ingresar al sistema deberá aparecer una segunda ventana “Detalle de Ventas” con un diseño parecido a este (No necesariamente igual):



Sobre la ventana, se deben considerar los siguientes criterios:

* En la tabla de la ventana deberá presentar todos los registros que se encuentren en la tabla detalle\_venta de la base de datos.
* El campo texto deberá permitirle al usuario buscar por el nombre del Articulo (la columna descripción en la base de datos). Los artículos con el mismo nombre deberán aparecer filtrados en la tabla de la ventana al presionar el botón Buscar.
* Al borrar el contenido del campo texto o al no ingresar nada en el campo. Y presionar el botón buscar, deberá mostrar en la tabla de la ventana todos los registros sin filtros.
* El botón Agregar deberá presentar un modal con el que el usuario deberá poder agregar un nuevo registro en la tabla detalle\_venta. El nuevo registro deberá mostrarse también en la tabla de la ventana.

Los campos de la tabla detalle\_venta son:

|  |  |  |
| --- | --- | --- |
| **Campo** | **Tipo de Dato** | **Observación** |
| Id | Int |  |
| Código | Varchar(10) |  |
| cantidad | Int |  |
| descripcion | Varchar(100) | Nombre del Artículo. |
| precio\_unitario | Money |  |
| precio\_total | Money |  |
| fecha | DateTime |  |

**Aplicativo en JAVA**

Se solicita la creación de un ejecutable JAVA, empaquetado en formato jar, que se pueda ejecutar por consola, la función de la aplicación será generar un archivo cada 5 minutos y registrar en la base de datos su generación.

El nombre del archivo debe formarse con la palabra ReporteVenta al inicio, guion bajo, luego la fecha actual en formato año mes día juntos, guion bajo, la hora actual en formato hora minutos segundos juntos. El formato del archivo debe ser txt. Ejemplo: ReporteVenta\_20200131\_183001.txt

En el archivo se deberá cargar la información de la tabla detalle\_venta, cada columna separada por un tab. Una vez creado, se deberá guardar el archivo en la misma ruta de la aplicación JAVA, en una carpeta archivos, fuera del src. La tabla en la que se debe registrar la creación del archivo con su ruta donde fue almacenada se llama reportes\_ventas.

Para la creación del archivo se debe considerar que:

* Cada archivo debe presentar la información de la tabla detalle\_venta ingresada desde la fecha de la última creación del archivo hasta la actual. Es decir 5 min de información ingresada.
* Si no existe un reporte\_venta creado anteriormente, entonces el archivo deberá considerar todos los detalle\_venta que se encuentren hasta la fecha actual.
* Se debe guardar en la tabla reporte\_venta, el detalle del archivo creado, junto con su fecha de creación, la que se debe usar como referencia de fecha inicio para la búsqueda de información que se agregara al nuevo archivo.
* La información de cada archivo no se puede repetir.
* Si no se ha registrado ningún nuevo registro en la tabla detalle\_venta, en el tiempo inicio fin, el archivo deberá generarse vacío.

Los campos de la tabla reporte\_venta son:

|  |  |  |
| --- | --- | --- |
| **Campo** | **Tipo de Dato** | **Observación** |
| Id | Int |  |
| nombre\_archivo | Varchar(33) |  |
| path | Varchar(MAX) | La ruta completa de la ubicación del archivo |
| fecha\_creation | DateTime |  |

**La prueba tiene una distribución en la calificación de la siguiente manera:**

|  |  |
| --- | --- |
| Distribución de Puntaje | Puntos |
| PHP: Creación de la Ventana de Autenticación | 1 Punto |
| PHP: Validación de Autenticación correcta | 1 Punto |
| PHP: Creación de Ventana de Detalle de Promociones | 1 Punto |
| PHP: Correcto funcionamiento de Filtro de Búsqueda | 1 Punto |
| PHP: Correcto funcionamiento de Formulario de Creación de nuevo detalle de venta | 1 Punto |
| JAVA: Creación de archivo en el formato establecido | 1 Punto |
| JAVA: Ciclo infinito de ejecución cada 5 min | 1 Punto |
| JAVA: Registro en la base de datos el detalle de la creación del archivo | 1 Punto |
| JAVA: Registro de información en el archivo sin repetirla | 2 Puntos |
| TOTAL | 10 Puntos |

Se recomienda un mínimo de 4 horas para la ejecución del test. Y un máximo de 8 horas para su entrega.